

Comparando Estruturas de Testes em uma Organização

Vanessa Cândido¹, Cidinha Gouveia²

¹C.E.S.A.R. – Centro de Estudos e Sistemas Avançados do Recife
Rua do Apolo, 81, Recife, PE, Brasil

² C.E.S.A.R. – Centro de Estudos e Sistemas Avançados do Recife
Rua do Apolo, 81, Recife, PE, Brasil

vanessa.freitas@cesar.org.br, cidinhacg@gmail.com

Abstract. *Rapid advances in technology have brought to the market the need of having more complex and critical software developed in a short period of time. Delivering high-quality software on-time and on-budget in this scenario is becoming a difficult challenge in IT companies. In order to minimize this problem faced by the market, IT companies are investing more in having a well defined software testing structure. However, lots of them fail in their first initiative because they do not know the existing structures and the risks involved in adopt each one. This paper will report an experience deploying three organizational structures at 4 C.E.S.A.R's project (Recife Center for Advanced Studies and Systems): Independent Test Teams, Integrated Test Teams and Outsourcing, telling how each structure was implemented, highlighting the difficulties, the benefits and it will provide a comparative study between the structures.*

Resumo. *Rápidos avanços na tecnologia têm trazido para o mercado a necessidade de sistemas de software mais complexos e de maior criticidade desenvolvidos em curto tempo. Entrega de software de qualidade dentro do tempo e prazo especificado no cenário atual de TI tem se tornado um grande desafio para as empresas de TI. Procurando minimizar este problema diante do mercado, empresas de TI estão dedicando mais tempo para a verificação e validação de seus produtos, investindo esforços para que suas estruturas de testes de software sejam bem definidas. Porém, um grande número dessas empresas falham na primeira iniciativa por não conhecerem os tipos de estruturas existentes e os riscos envolvidos em cada uma delas. Este artigo relatará uma experiência na implantação de três estruturas organizacionais de testes: Equipe Independente de Testes, Equipe Integrada de Testes e Terceirização, em quatro projetos do C.E.S.A.R.(Centro de Estudos e Serviços Avançados do Recife), descrevendo como cada estrutura foi implementada, enfatizando as dificuldades, os benefícios encontrados e fornecendo um comparativo entre as três estruturas.*

1. Introdução

Cada vez mais, as empresas estão se conscientizando que entregar *software* de alta qualidade dentro do prazo e do orçamento previamente acordado tem se tornado um grande desafio. As funcionalidades dos sistemas de *software* estão aumentando em

complexidade, tamanho, criticidade e o tempo de desenvolvimento está sendo reduzido. Todos esses fatores reunidos significa um grande risco para a qualidade do *software* e aumenta a probabilidade do não cumprimento do prazo e do custo acordado Mjos e Veritas (2006). De acordo com uma pesquisa realizada pelo silicom.com Jain (2007), um em cada três projetos de tecnologia ultrapassa o seu orçamento e um em cada quatro projetos custa 50% a mais do que o esperado.

A necessidade de melhorar a qualidade do *software* e executar um projeto dentro dos custos e tempo despertou nas empresas de TI, uma busca pela necessidade de ter uma estrutura bem definida de testes. Muitas empresas falham na primeira iniciativa de realizar as atividades de testes porque não conhecem os tipos de estruturas organizacionais de testes existentes, nem os riscos envolvidos em cada uma e a forma de tratá-los. Como mencionado em alguns livros Black (2003), Craig and Jaskiel (2002), há muitos caminhos de organizar as atividades de testes, tais como: Equipe Independente de Testes, Equipe Integrada de Testes, Desenvolvedores que desempenham o papel dos testadores, Terceirização, Coordenador de Testes, etc.

Este artigo relatará uma experiência na implantação de três estruturas organizacionais de testes em quatro projetos de uma organização, e realizará também uma comparação entre as mesmas. As estruturas implantadas foram: Equipe Independente de Testes, Equipes Integrada de Testes e Terceirização.

A seção 2 apresentará os itens de análise que foram usados para o relato deste trabalho, a seção 3 detalhará a experiência na implantação de cada estrutura em quatro projetos do C.E.S.A.R, uma empresa de inovação baseada em projetos de TI, a seção 4 fornecerá um estudo comparativo entre os resultados dos projetos. A seção 5 apresentará as conclusões deste trabalho.

2. Itens de Análise

Baseado na experiência da equipe, cinco itens, primordiais para o sucesso da atividade de teste, foram escolhidos para análise:

Comunicação: A maioria dos livros de testes dedicam diversas páginas para questões técnicas do processo de testes Black (2003), Craig e Jaskiel (2002) e Dustin (2002). Contudo, muitas iniciativas de testes falham devido à má comunicação entre as pessoas envolvidas. Desenvolvedores tendem a pensar que o trabalho dos testadores reduz a velocidade dos esforços do desenvolvimento e terminam mantendo uma postura distante. Este problema abre várias questões, tais como: análises e correções demoradas das falhas, divergências de opiniões entre desenvolvedores e testadores a respeito dos *bugs* encontrados, etc. Os gerentes precisam promover um bom relacionamento e uma boa comunicação dentro do time.

Testes no início do ciclo de desenvolvimento: Quanto mais rápido as atividades de testes iniciarem, mais barato é o custo de correção dos erros encontrados. A remoção de um erro pode custar 100 vezes mais se ele for encontrado numa fase mais tarde do ciclo de desenvolvimento Dustin (2002). Por isso, a necessidade de começar os testes, tão rápido quanto o possível é primordial.

Nível de Independência: Na maioria das vezes, os marcos do projeto parecem ser algo imutável. A pressão para que o produto seja entregue dentro do prazo estabelecido

pode se tornar tão alta que alguns gerentes tendem a ignorar os resultados das atividades de testes para não atrasar a liberação do produto. Se a equipe de testes tiver um bom nível de independência, influenciando nas decisões do projeto, o risco de liberar um produto ruim é bastante reduzido. Se a equipe de testes tiver um nível de independência maior aumenta as chances de sucesso das atividades de testes e com isso melhora a qualidade do produto Test Maturity Model (2008).

Nível de Especialização do Profissional: A maioria dos cursos de Ciência da Computação não tem uma disciplina de Testes e não prepara os alunos para o mercado na área de Testes. Pouco a pouco, algumas universidades estão oferecendo cursos específicos na área de Testes, mas a dificuldade de encontrar bons profissionais no mercado é ainda um problema. A especialização dentro da área de Testes de *Software* depende fortemente de uma experiência prática, o que pode ser influenciada pelo tipo de estrutura organizacional escolhida.

Custos: O problema com a avaliação dos custos dos testes para a maioria das organizações é que não é conhecido o custo atual dos testes, e também não são percebidos os custos dos testes inadequados Hetzel (1988). Neste artigo relataremos os custos envolvidos na implementação de cada tipo de estrutura.

3. Implementando diferentes estruturas de testes nos projetos do C.E.S.A.R.

3.1 Terceirização

Terceirização é uma forma de melhorar a qualidade do produto de *software*, na qual uma empresa independente da organização desenvolvedora é contratada para realizar verificação e validação do produto, conforme ilustrado na Figura 1. Dois projetos utilizaram a estrutura de Terceirização, sendo um no qual o C.E.S.A.R. foi a empresa contratada para realizar as atividades de testes e outro no qual o C.E.S.A.R. foi o contratante dos serviços de testes.

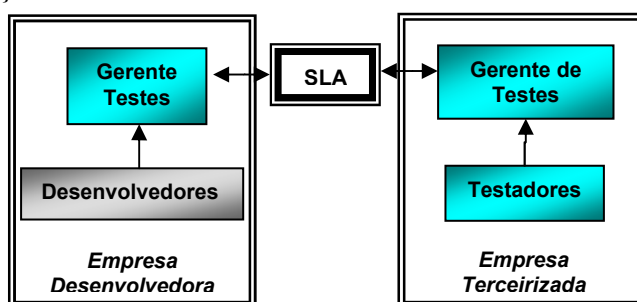


Figura 1 - Estrutura de Terceirização

3.1.1 Projeto A - Terceirização - C.E.S.A.R. como um contratado

Neste projeto, o C.E.S.A.R. exerceu o papel de um contratado para oferecer serviços de testes. Cerca de 100 engenheiros de testes foram envolvidos ao longo de quatro anos. A comunicação entre os testadores e os desenvolvedores foi realmente difícil no começo, especialmente porque os desenvolvedores estavam afastados ao redor do mundo, mas este problema foi mitigado com algumas ações, tais como: frequentes viagens em visita ao local de trabalho dos desenvolvedores e vice-versa; reuniões semanais para discussões sobre os bugs encontrados, inspeções de requisitos e de código, uso de ferramentas de comunicação como Messenger, e-mails, Netmeeting, Skype. Depois

destas ações a comunicação melhorou significativamente. Foi verificado que quando o contato é mantido de forma satisfatória, o relacionamento entre as pessoas torna-se melhor, a confiança aumenta e o trabalho é realizado de forma mais rápida e com uma melhor qualidade.

Os testes sempre foram feitos desde o início do processo de desenvolvimento; isto foi fácil de ser realizado porque a empresa desenvolvedora do *software* já tinha consciência desta necessidade.

Como definido no SLA (Service Level Agreement), as regras foram bastante rígidas e a equipe desenvolvedora sempre teve um forte acompanhamento do projeto. Eles estiveram presentes em quase todas as reuniões técnicas, sempre mostrando interesse nos resultados dos testes e ouvindo a nossa opinião, nossos interesses, nossas dúvidas, procurando solucionar os problemas e tomar algumas decisões em conjunto. Os testes tiveram uma completa independência e que os resultados dos testes tiveram total influência nas estratégias e decisões do time de desenvolvimento.

Prover testes como um serviço contribuí para uma melhor especialização dos recursos em áreas específicas, pois o formato dessa estrutura possibilita a criação de grupos focados na parte de design de testes, execução de testes, automação de testes, testes exploratórios, etc. Além do trabalho que eles estão habituados a fazer, eles também costumam sugerir e implementar melhorias nas suas atividades, gerando novas estratégias, metodologias e ferramentas, tornando o processo de testes mais eficiente e efetivo. Algumas dessas melhorias e seus resultados foram publicados por colaboradores do C.E.S.A.R., tais como: Gouveia, Oliveira e Quidute (2006), Lima e Bandeira (2007), Assad (2005).

Medindo o número de *bugs* encontrados, sua criticidade e o tempo gasto para que o mesmo fosse encontrado, foi possível levantarmos o quanto podemos economizar para a empresa desenvolvedora do *software* e contribuir para aumentar a qualidade do produto. Isto também foi fortalecido pelo feedback do cliente.

3.1.2 Projeto B - Terceirização – C.E.S.A.R. como um contratante

Este projeto foi executado utilizando estrutura de terceirização, mas desta vez o C.E.S.A.R. assumiu o papel de contratante das atividades de testes. Neste projeto tivemos 3 engenheiros de testes vindos da empresa contratada realizando as atividades de testes durante um período de 4 a 6 meses. Como ambas as empresas estão localizadas na mesma cidade, para tornar a comunicação mais eficiente, a empresa contratante requisitou a empresa contratada que os engenheiros de testes dela fossem enviados para trabalhar fisicamente próximo aos desenvolvedores da empresa contratante. Isto foi uma boa ação para melhorar o link entre testadores e desenvolvedores e conseqüentemente a comunicação. Tecnicamente também funcionou bem. Contudo, outros tipos de impactos foram detectados. A empresa contratada enviou os engenheiros de testes, mas não enviou o gerente de testes porque ele também era responsável por outros projetos e devido a isto não poderia está fisicamente alocado fora da sua empresa. Houve uma perda do gerenciamento dos testadores pela parte do gerente de testes da empresa terceirizada, então com isso eles terminaram sendo influenciados pelo processo do C.E.S.A.R. e controlados pelo gerente de projeto do C.E.S.A.R., causando diversas divergências, impactando bastante nas atividades de testes. Por um lado, não houve

dificuldades de começar a execução das atividades de testes no início do ciclo de desenvolvimento do projeto, uma vez que todos estavam alinhados em relação à importância dos testes desde esse momento. Por outro lado, como o tempo de conclusão do projeto foi bastante curto, o SLA não foi bem definido nem aplicado corretamente no projeto. Devido ao projeto ter sido curto, o gerente de projeto teve dificuldades em realizar um adequado acompanhamento com o time de testes. Os resultados providos foram sempre analisados pelo gerente do projeto do C.E.S.A.R e eles costumavam tomar decisões sem a influência do gerente de testes da empresa contratada. Neste tipo de estrutura, espera-se um bom nível de independência, mas não houve neste projeto. Na prática, o gerente de testes não contribuiu com as decisões dos projetos.

Usualmente, numa estrutura de terceirização tem-se melhores chances de conseguir recursos especializados. Contudo, as preocupações com a perda de comunicação foi muito maior do que o fato de não ter recursos especializados. A ação de trazer alguns testadores da equipe contratada para ficarem próximos dos desenvolvedores atrapalhou o melhor uso de recursos especializados da empresa contratada, visto que nossos recursos foram fixos e não houve devido a isto, a oportunidade de trabalhar com outros recursos mais especializados.

3.2 Projeto C - Equipe Integrada de Testes

Equipes Integrada de Testes são equipes formadas por desenvolvedores e testadores que reportam a mesma gerência Craig e Jaskiel (2002), como mostrado na Figura 2.

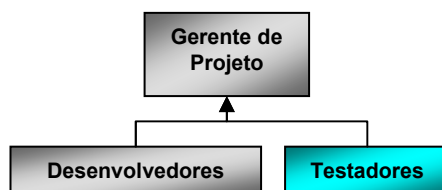


Figura 2 - Estrutura de Equipe Integrada de Testes

A maioria dos projetos do C.E.S.A.R, atualmente, usam a estrutura de equipe integrada de testes, e um dos projetos que escolhemos para ser relatado neste trabalho foi um projeto com as seguintes características: curta duração (4-6 meses), número de profissionais envolvidos entre 8 e 15 e número de testadores entre 3 e 4. A comunicação nesse projeto foi muito satisfatória. Os times trabalharam bastante conectados e o projeto realmente teve um andamento rápido.

Neste projeto, tivemos também, a facilidade de iniciar os testes desde o início do ciclo de desenvolvimento do *software*.

Toda a equipe foi formada por profissionais do C.E.S.A.R., os quais já estavam acostumados com os processos, ferramentas e estratégias da empresa. Isto ajudou bastante para que as ações necessárias fossem realizadas de forma rápida.

Usualmente, esta estrutura tende a apresentar um baixo nível de independência dos testes, mas não aconteceu nesse projeto. Acreditamos que foi devido ao gerente do projeto já conhecer o time e a forma deles trabalharem e pelo fato da equipe já estar familiarizada com os processos e ferramentas da empresa, isto proporcionou um bom relacionamento de confiança. O gerente costumava ouvir a opinião da equipe de testes antes de tomar certas decisões. Então, podemos concluir que a equipe de testes teve um

alto nível de independência influenciando realmente o gerente de projeto através dos resultados dos testes e das orientações, quando necessárias, da equipe de testes.

Infelizmente, não houve tempo para especialização dos recursos uma vez que o projeto foi curto e houveram diversas atividades de testes a serem realizadas em um pequeno período de tempo.

Comparando com a estrutura de terceirização, onde as atividades de testes foram contratadas, os custos utilizando essa estrutura foram bem mais baixos e a qualidade dos serviços que esperávamos foi alcançada. De acordo com o *feedback* do gerente do projeto e do cliente, o número de *bugs* encontrados depois da entrega do produto foi bastante baixo (comparado com um projeto que utilizou terceirização e com a média de outros projetos do C.E.S.A.R.) e eles não foram *bugs* críticos.

3.3 Projeto D - Equipe Independente de Testes – GRIT

Uma Equipe Independente de Testes é um time no qual o objetivo principal do trabalho são as atividades de Testes. Eles podem trabalhar com apenas um produto ou vários. O gerente de testes não reporta para o gerente de desenvolvimento e devem está no mesmo nível hierárquico dentro da organização Craig e Jaskiel (2002), como mostrado na Figura 3.

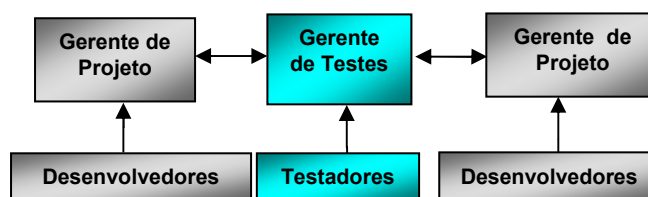


Figura 3 - Estrutura de Equipe Independente de Testes

Mesmo com a estrutura de equipe integrada de testes oferecendo bons resultados, o C.E.S.A.R. ainda tinha o problema de não ter recursos especializados em testes. Cada engenheiro de teste estava alocado em um projeto. Eles não trocavam idéias nem experiências vivenciadas em seus projetos, não ajudando dessa forma para que os mesmos tornarem-se mais *experts*. Dentro deste contexto, foi detectada a necessidade de guiar e monitorar as atividades de testes dentro da organização. Isto foi a razão para a criação do GRIT (Grupo Independente de Testes). Este grupo é baseado em uma estrutura de equipe independente de testes. Ele provê serviços para toda a empresa. Contudo, eles são serviços gerais, tais como: acompanhar como as atividades de testes estão sendo realizadas dentro de cada projeto; provê um processo de teste geral que possa ser utilizado em todos os projetos e mantê-lo atualizado; oferecer ferramentas para ajudar na execução das atividades de testes e na análise de métricas de testes dos projetos e da empresa; sugerir ações em alguns projetos caso seja necessário; oferecer treinamentos para os novos engenheiros de testes; oferecer orientações para pesquisas e produções científicas na área; ajudar gerentes de projeto no processo de estimativa de testes; provê seminários de testes na empresa e promover a cultura de testes. Contudo, o GRIT não elimina a necessidade de ter engenheiros de testes alocados diretamente nos projetos específicos.

O GRIT provê suporte para a área de testes dentro do C.E.S.A.R., dando direções corretas e funciona como uma sistematização da área. O grupo é formado por sete áreas, onde cada uma é responsável por um tipo de serviço. O gerente de testes não

pode tomar decisões dentro de cada projeto, mas ele pode influenciar bastante, uma vez que ele é responsável pelo controle da área de testes da organização. Este tipo de estrutura tem se mostrado bastante eficiente. O gerente de testes tendo uma boa influência nas decisões dos projetos permite que os testadores sentem-se mais confortáveis com o trabalho deles.

A boa comunicação foi extremamente necessária para o sucesso do grupo. Os resultados do GRIT são frequentemente apresentados para a empresa. Os seus resultados podem ser vistos na prática e com isso podemos ganhar confiança sobre os trabalhos realizados com as atividades de testes.

O desempenho do GRIT reforça a consciência de um maior número de pessoas na organização sobre a importância dos testes, facilitando o início dos testes o mais rápido possível durante o ciclo de desenvolvimento de um *software*, uma vez que os gerentes de projeto começam a se preocupar mais sobre esta questão.

O grupo é completamente independente dos projetos da organização. Isto ajuda bastante a velocidade do seu trabalho, pois não depende de aprovações de nenhum projeto específico.

Profissionais que participam do GRIT tendem a ser mais experts uma vez que eles vivenciam diferentes tipos de problemas. Células especializadas de serviços específicos foram criadas para ajudar na solução desses problemas

É um dos melhores pontos a serem considerados é o fato de que o GRIT não necessita de custos adicionais para a organização. Os engenheiros de testes que formam o grupo também estão alocados em outros projetos, nos quais participam das atividades de sua responsabilidade e em paralelo realizam as suas tarefas do GRIT. Eles costumam ser mais produtivos em seus projetos devido à troca de experiências com os outros membros do GRIT.

4. Análise das estruturas de testes

A Tabela 1 mostra um resumo de uma comparação dos projetos nos quais foram implantadas as diferentes estruturas, baseado no nível de tendência das variáveis.

Tabela 1 – Tendências das Variáveis em cada Projeto

Variáveis	Projetos - Estruturas de Equipes de Testes			
	<u>PROJETO A</u> <i>Terceirização - C.E.S.A.R. como um contratado</i>	<u>PROJETO B</u> <i>Terceirização - C.E.S.A.R. como um contratante</i>	<u>PROJETO C</u> <i>Equipe Integrada de Testes</i>	<u>PROJETO D</u> <i>Equipe Independente de Testes - GRIT</i>
Comunicação	Ruim	Médio	Excelente	Bom
Testes no início do ciclo de desenvolvimento	Moderado	Difícil	Fácil	Moderado
Nível de Independência	Alto	Baixo	Alto	Médio
Nível de Especialização dos Profissionais	Alto	Baixo	Baixo	Médio
Custos	Alto	Alto	Baixo	Médio

5. Conclusão

Neste trabalho, relatamos três diferentes tipos de estruturas aplicadas em quatro projetos do C.E.S.A.R., explicando os benefícios alcançados, dificuldades encontradas e os resultados de cada tipo de modelo.

Realmente não há uma forma certa ou errada de se organizar a estrutura de testes em uma organização, na realidade a maioria das estruturas funciona em algumas empresas e falham em outras Craig e Jaskiel (2002). Para escolher a melhor estrutura para a sua empresa ou a combinação delas, é importante compreender o comportamento de algumas variáveis dentro de cada tipo de estrutura de testes existente; conhecer bem a operação da organização e seus profissionais e saber os riscos envolvidos em cada tipo de estrutura.

Depois de vivenciar diferentes tipos de problemas, o C.E.S.A.R. escolheu utilizar uma estrutura híbrida de testes, para que fosse possível conseguir o melhor custo-benefício para a empresa. Uma equipe integrada de testes, uma vez que temos engenheiros de testes alocados em nossos projetos respondendo diretamente ao gerente de projeto e uma equipe independente de testes para oferecer suporte a algumas deficiências do outro modelo usado. Depois de 4 anos testando diferentes tipos de estruturas ao mesmo tempo tentando mudar a cultura da organização, ainda estamos procurando sempre realizar melhorias. Profissionais mudam de uma empresa para outra, novos processos e novas tecnologias são implementadas e sempre estamos reavaliando o nosso modelo para checar se ele ainda se adequa a atual realidade do C.E.S.A.R.

6. Referências

- Black, R. (2003). Critical testing processes: plan, prepare, perform, perfect.
- Craig, R. D., Jaskiel, S. P. (2002) Systematic *Software* Testing.
- Test Maturity Model. (2008) Um guia para usar o modelo TMM, provido pelo website Fundação TMM, disponível em: <http://www.tmmifoundation.org>
- Jain, B.(2007) One in three tech projects runs over budget, disponível em: <http://management.silicon.com/itdirector/0,39024673,39168429,00.htm>
- Mjos, N. Veritas, D. N. (2006) Independent *Software* Verification & Validation Guide, *Software & Systems Quality Conferences*, Alemanha.
- Dustin, E., (2002). Effective *Software* Testing, Addison Wesley.
- Gouveia C., Oliveira J., Quidute R.(2006) A Way of Improving Test Automation Cost-Effectiveness, CAST – Conference Association for *Software* Testing.
- Lima P. O., Bandeira L. R (2007)., Boas Práticas Adotadas em um Projeto de Design de Testes, II Encontro Brasileiro de Teste de *Software*.
- Assad A.(2005) OSGi Applications Testing, OSGi Alliance Mundial Congress, Paris, França.
- Hetzel B. (1988) The Complete Guide to *Software* Testing, segunda edição, QED Ciência de Informações Sciences Inc., Wellesley, MA.